

WHAT IS CLAIMED IS:

- 1 1. A method for testing code, comprising:
2 receiving a test request;
3 sending executable program code, corresponding to the
4 test request, to a client system;
5 receiving a response from the client system indicating
6 that the client system will perform a test, and
7 indicating that the client system was not being
8 actively used when the executable program code
9 was sent.
- 1 2. The method of claim 1, wherein executable program
2 code, corresponding to the test request, is sent to
3 multiple client systems.
- 1 3. The method of claim 1, further comprising retrieving a
2 list of client system identifiers, the client system
3 identifiers indicating client systems to which
4 executable program code can be sent for testing.

- 1 4. A method for testing code in a client data processing
2 system, comprising:
3 receiving executable code from a server system in a
4 client data processing system;
5 if the client data processing system is in an idle
6 state when the executable code is received, then
7 sending a response to the server system,
8 testing at least a portion of the executable
9 code, and
10 sending test results to the server system.
- 1 5. The method of claim 4, wherein if the client data
2 processing system is not in an idle state when the
3 executable code is received, then no response is sent
4 to the server and no testing is performed.
- 1 6. The method of claim 4, wherein the testing is a
2 coverage analysis test.
- 1 7. The method of claim 4, wherein the client data
2 processing system is in an idle state when no user is
3 actively operating the client data processing system.

1 8. A data processing system having at least a processor
2 and accessible memory, comprising:
3 means for receiving a test request;
4 means for sending executable program code,
5 corresponding to the test request, to a client
6 system;
7 means for receiving a response from the client system
8 indicating that the client system will perform a
9 test, and indicating that the client system was
10 not being actively used when the executable
11 program code was sent.

1 9. The data processing system of claim 8, wherein
2 executable program code, corresponding to the test
3 request, is sent to multiple client systems.

1 10. The data processing system of claim 8, further
2 comprising means for retrieving a list of client
3 system identifiers, the client system identifiers
4 indicating client systems to which executable program
5 code can be sent for testing.

1 11. A data processing system having at least a processor
2 and accessible memory, comprising:
3 means for receiving executable code from a server
4 system in a client data processing system;
5 means for, if the client data processing system is in
6 an idle state when the executable code is
7 received,
8 sending a response to the server system,
9 testing at least a portion of the executable
10 code, and
11 sending test results to the server system.

1 12. The data processing system of claim 11, wherein if the
2 client data processing system is not in an idle state
3 when the executable code is received, then no response
4 is sent to the server and no testing is performed.

1 13. The data processing system of claim 11, wherein the
2 testing is a coverage analysis test.

1 14. The data processing system of claim 11, wherein the
2 client data processing system is in an idle state when
3 no user is actively operating the client data
4 processing system.

1 15. A computer program product tangibly embodied in a
2 machine-readable medium, comprising:
3 instructions for receiving a test request;
4 instructions for sending executable program code,
5 corresponding to the test request, to a client
6 system;
7 instructions for receiving a response from the client
8 system indicating that the client system will
9 perform a test, and indicating that the client
10 system was not being actively used when the
11 executable program code was sent.

1 16. The computer program product of claim 15, wherein
2 executable program code, corresponding to the test
3 request, is sent to multiple client systems.

1 17. The computer program product of claim 15, further
2 comprising instructions for retrieving a list of
3 client system identifiers, the client system
4 identifiers indicating client systems to which
5 executable program code can be sent for testing.

1 18. A computer program product tangibly embodied in a
2 machine-readable medium, comprising:
3 instructions for receiving executable code from a
4 server system in a client data processing system;
5 instructions for, if the client data processing system
6 is in an idle state when the executable code is
7 received,
8 sending a response to the server system,
9 testing at least a portion of the executable
10 code, and
11 sending test results to the server system.

1 19. The computer program product of claim 18, wherein if
2 the client data processing system is not in an idle
3 state when the executable code is received, then no
4 response is sent to the server and no testing is
5 performed.

1 20. The computer program product of claim 18, wherein the
2 testing is a coverage analysis test.

1 21. The computer program product of claim 18, wherein the
2 client data processing system is in an idle state when
3 no user is actively operating the client data
4 processing system.